Sub-Meter Vehicle Navigation Using Efficient Pre-Mapped Visual Landmarks

Han-Pang Chiu, Mikhail Sizintsev, Xun S. Zhou, Philip Miller
Supun Samarasekera, Rakesh (Teddy) Kumar

***Abstract*— This paper presents a vehicle navigation system that is capable of achieving sub-meter GPS-denied navigation accuracy in large-scale urban environments, using pre-mapped visual landmarks. Our navigation system tightly couples IMU data with local feature track measurements, and fuses each observation of a pre-mapped visual landmark as a single global measurement. This approach propagates precise 3D global pose estimates for longer periods. Our mapping pipeline leverages a dual-layer architecture to construct high-quality pre-mapped visual landmarks in real time. Experimental results demonstrate that our approach provides sub-meter GPS-denied navigation solutions in large-scale urban scenarios.**

I. Introduction

Accurate estimation of 3D global position of a moving vehicle in a continuous manner is crucial to future driver assistance systems or autonomous driving applications [6]. Sub-meter level precision is necessary for situations such as obstacle avoidance or predictive emergency braking. The typical solution to achieve this accuracy is to fuse high precision differential GPS with high-end inertial measurement units (IMUs), which is prohibitively expensive for commercial purpose. Non-differential GPS can be cheap, but rarely reach satisfactory accuracy in urban environments due to signal obstructions or multipath effects.

To solve this problem, localization methods using a pre-built map of visual landmarks have received lots of attention in recent years. These methods typically require a stereo camera carefully calibrated with GPS receiver for map building. The map of the environment is constructed and geo-referenced beforehand, and is used for global positioning during future navigation by matching new feature observations from on-board perception sensors to this map.

However, these methods typically obtain one single pose estimate from all visual landmark associations at a given time to update vehicle pose state estimate during navigation. This way only achieves decent overall accuracy if there are sufficient mapped landmarks continuously matched to high-frequency perceived images all the way during driving. That is difficult when the vehicle is occluded in traffic scenes or the scene appearance is changed by differences in lighting or weather variation. Moreover, the mapping process of these methods is time-consuming, which take hours or days after data collection to reconstruct large-scale 3D scene points using bundle adjustment optimization [20].

The authors are with Center for Vision Technologies, SRI International, Princeton, NJ 08540, USA. The contact author is Han-Pang Chiu {han-pang.chiu@sri.com}

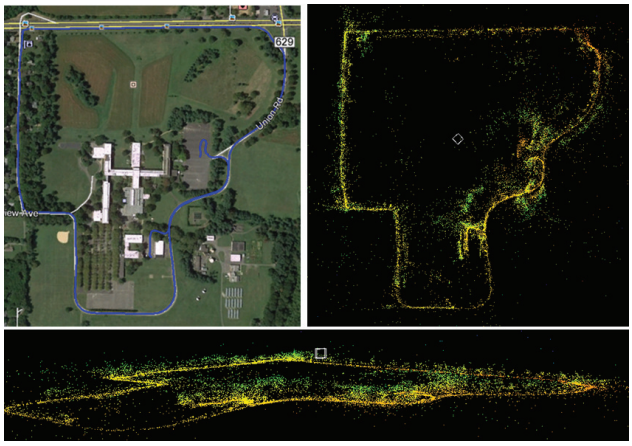

Fig. 1: 3D visual landmarks mapped around our campus in real time: (top left) Google earth view of the mapped trajectory, (top right) Top view of mapped landmarks in color point cloud representation, (bottom) Side view of mapped landmarks in color point cloud representation. The color code reflects the absolute height, and shows landmarks correspondent to road and trees. The small white box shows the 3D perspective viewpoint.

In this paper, we show that sub-meter vehicle navigation accuracy in large-scale urban environments can be achieved without GPS by efficiently using pre-mapped visual landmarks (Figure 1). Our navigation system continuously estimates 3D global pose by tightly fusing IMU data, local tracked features from a monocular camera, and global landmark points from associations between new observed features and pre-mapped visual landmarks. Unlike previous methods, we treat each new observation of a pre-mapped visual landmark as a single measurement instead of computing only one pose measurement from all landmark observations at a given time. This way tightly incorporates geo-referenced information into landmark measurements, and is capable of propagating precise 3D global pose estimates for longer periods in GPS-denied setting. It is also more robust to places where only few or no pre-mapped landmarks are available due to scene occlusion or changes.

The high-quality map of visual landmarks used for our navigation system is constructed beforehand by using a monocular camera, IMU, and high-precision differential GPS. Our mapping process leverages recent advances in real-time large-scale Simultaneous Localization and Mapping (SLAM) works which utilize a dual-layer architecture: low-latency navigation updates provide an initial estimate for slower map optimization running in parallel. When our data collection driving is finished, the fully optimized map is

immediately available for future use.

We summarize our contributions in this paper as follows.

- A GPS-denied navigation system which efficiently utilizes pre-mapped visual landmarks to achieve sub-meter overall global accuracy in large-scale urban environments using only IMU and a monocular camera.
- A real-time process to build a high-quality, fully-optimized map of visual landmarks using IMU, GPS, and one monocular camera.

The remainder of this paper begins with a discussion of related work in Section II. Section III introduces our mapping process, which includes how we select 2D features from key video frames and optimize the estimation of the associated 3D landmarks in the map. Section IV describes our high-precision GPS-denied navigation system, including how we formulate individual landmark observations, which associate new camera features to pre-mapped visual landmarks, in our sensor fusion framework. We demonstrate our approach provides sub-meter GPS-denied navigation solutions in large-scale urban scenarios in Section V followed by our conclusions in Section VI.

II. Related Work

In recent years, many visual SLAM works [5] have been proposed for vehicle navigation, which try to solve the problem by mapping an unknown area while localizing the vehicle within the map at the same time. By using a parallel architecture [8], [16], [19], it can achieve real-time navigation updates with slower map optimization running in parallel to process expensive loop closures. The navigation error will increase until the vehicle re-visits the mapped area. There are also visual-inertial navigation works [3], [11] which integrate IMU data with feature track measurements to further reduce the drift rate. However, without absolute measurements, none of these methods maintain overall sub-meter global accuracy within large-scale urban environments.

The availability of a pre-optimized map is able to simplify the vehicle navigation problem as pure localization, without the need for simultaneous map construction and loop closure detection. Instead of using costly and bulky sensors such as laser scanners [6], [12], there are a number of methods [1], [10], [21], [22] that propose to use cameras to construct a geo-referenced map of visual landmarks beforehand. 2D feature points with descriptors are detected and extracted from multiple images to reconstruct 3D points using bundle adjustment optimization. Each optimized visual landmark in the map consists its absolute 3D coordinate, with its 2D locations and visual descriptors from 2D images. This map of 2D-3D visual landmarks then provides absolute measurements for future vehicle navigation. For example, Lategahn et al. [10] builds an optimized map of 3D landmarks and their 2D descriptors using GPS and stereo camera. The map is then used to localize a vehicle with a monocular camera. The localized pose relative to the map is refined with IMU measurements.

To achieve desired accuracy during navigation, these methods rely on continuous and sufficient matches between new visual features and pre-mapped landmarks. They are not durable to situations when landmarks are not available such as the vehicle is occluded in traffic scenes or the scene appearance has been changed. In addition, the mapping process of these methods typically requires at least hours of work to reconstruct enormous number of landmarks using bundle adjustment process, after data collection.

There are three major differences between our work and previous methods using pre-mapped visual landmarks. First, we adopt a tightly-coupled approach [14] to fuse IMU data and visual feature measurements through non-linear optimization for both our map building process and navigation system. Unlike the loosely-coupled method in [10] which ignores correlations among internal states from IMU and vision sensors during navigation, our approach models IMU error propagation as system dynamics to fuse individual feature track measurements from cameras. It is more amenable to nonlinear optimization, and has a lower drift rate without absolute measurements as shown in [11].

Second, our navigation system focuses on how to fully integrate pre-mapped landmark observations into a probabilistic sensor fusion framework. We treat each observation of a pre-mapped landmark as a single measurement instead of computing only one pose measurement [1], [10] from all landmark observations at a given time. This way allows finer level of individual landmark modeling with different uncertainties, and tightly incorporates absolute geo-referenced information into measurements for propagating precise 3D global pose estimates for longer periods.

Third, we propose a real-time mapping process by leveraging a parallel visual-inertial SLAM architecture inspired from [3], [16], [19]. Differential GPS measurements are also used in the fusion process to improve the quality of map construction. Based on the uncertainty of landmark states through non-linear optimization, only high-quality visual landmarks are selected and fully optimized in the map for future use.

In addition, we are more interested in overall global navigation accuracy including places where only few or no pre-mapped landmarks are available due to scene occlusion or appearance changes. Therefore, instead of calculating the relative distance to the visual map as the localization error [10], we evaluate our GPS-denied navigation accuracy using solutions provided by fusing high-precision differential GPS with IMUs.

III. Real-Time Geo-Referenced Map Building

In this section, we introduce our real-time mapping system (Figure 2) which fuses measurements from one monocular camera, IMU, and high-precision differential GPS. The map stores visual landmarks, which consists of 3D estimates (estimated from inference engine) with associated 2D information (position and descriptor from landmark matching module). To build this map, data from a monocular camera must be processed to extract meaningful measurements for our inference engine. Here we first introduce our visual odometry module which tracks features from previous frame

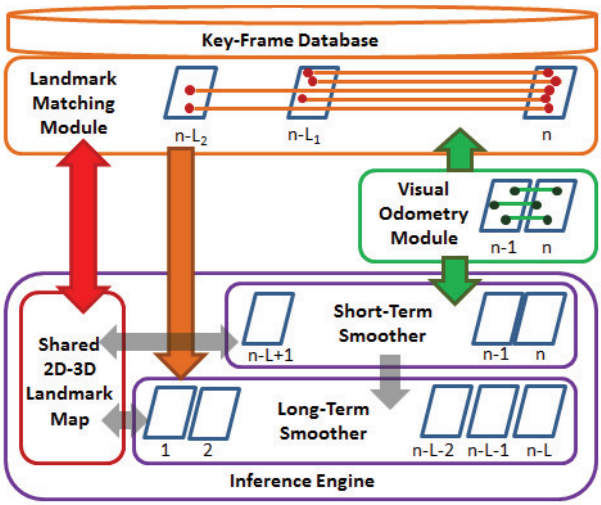

Fig. 2: The architecture of our real-time mapping system. The visual odometry module tracks features from previous frame to current frame for the short-term smoother. The landmark matching module selects and stores 2D features from images as key frames, and identifies loop closures by checking feature matches between past key frames and current image. Loop closures are utilized in the long-term smoother for slower map optimization.

to current frame. We will also present our landmark matching module. It constructs a database of key frames of selected 2D features. This module also queries and searches the database, and checks whether it can find associations between the new coming video frame and the past key frames. The identified associations form loop closures to further optimize the involved 3D landmarks in the map.

A. Visual Odometry

The visual odometry module efficiently associates sequential video frames from a monocular camera. It detects and matches features across consecutive frames, and also rejects outliers using additional rigid motion constraints (details in Section IV-B). All valid tracked features then become measurements generated from the module.

To find the balance between computation time and tracking performance, we have evaluated many choices of feature detectors and descriptors. Currently we use a slightly modified version of Harris corner detector [7] where an image is subdivided into tiles (e.g. 64×48 for 640×480 image). The strongest 10 corners are chosen in each tile, This way provides a uniform and dense spread of feature points extracted from a single video frame. Furthermore, they are extracted at 3 levels of Gaussian pyramid built from original image to cover multiple scales. Note this step is important for the landmark matching module described in Section III-B, which matches images captured from different viewpoints or from different cameras with varied intrinsic calibration parameters. We use the BRIEF descriptor [2], which is very fast to compute and correlate, to match detected features from previous frame to current frame.

Currently our average processing time for the entire process takes only 15 milliseconds to process around 850 features for image size of 640 pixels by 480 pixels (including outlier rejection steps described in Section IV-B), using a single core of an Intel i7 CPU running at 2.80 GHz.

B. Landmark Matching

The landmark matching module constructs a database of past key frames of visual features, and matches features between the query frame and past key frames in the database. It establishes feature associations across non-consecutive frames taken at different times.

1) Database Search : For each input frame, its extracted features are passed from the visual odometry module to search the database. For visual features, we re-compute the HOG descriptors [4] for the Harris corners detected in the visual odometry module. Compared to BRIEF descriptors used in visual odometry module, HOG descriptors are slower to compute and match. However, due to richer descriptor structure and rotation invariance, they perform significantly better than BRIEF descriptors when matching across non-sequential frames taken from different viewpoints, which is a more challenging task than matching across sequential frames.

Then we perform a self-matching test to select only distinctive features to search the database. For each feature on the input frame, this test matches the feature itself to all features on the frame. It finds the best match, which would be the feature itself since descriptors are identical, and second best match. The second best match error will be high if the feature is distinctive. Using a simple threshold on the second best match error allows us to remove non-distinctive features from the input frame. This way improves the performance of the entire search process, also reduces the computation time and database size, which is proportional to the number of features. For example, in case of 128-byte HOG and L2 match metric, the threshold of 200 results in good balance of feature distinctiveness and quantity.

Our database search mechanism is based on efficient Fast Library for Approximate Nearest Neighbors (FLANN) which searches in high-dimensional tree structure (randomized KD-forest, a collection of KD-trees) [15]. It finds K best feature matches which vote possible keyframe candidates. All valid feature associations from the keyframe candidate with the most inliers serve as final measurements generated from landmark matching module.

2) Database Construction : The query frame can also be selected as a new key frame and be added into the database. Note that database entry is essentially a keyframe that holds the collection of keypoints with their descriptors, image locations, and 3D world coordinates computed from triangulation across matched 2d points across frames. Thus, size of the database essentially depends on the number of keyframes, number of points, and matching descriptor in use. Various strategies to keyframe selection were explored. A simple strategy is to add the keyframe based on a fixed time interval (such as every 1 or 0.5 seconds) or spatial interval (such as every traveled 1 or 3 meters). A more advanced strategy has been adapted from [13], where the selection is based on conditions between query frame and

past key frames, including the number of overlapped features, the temporal difference, and the spatial difference between poses associated with frames. The current rules of thumb to add visually-representative (also diverse) key frames are the number of overlapped features should be small (e.g. 5)and the temporal difference should be large. The advantage of this strategy is that it limits the growth of database especially when driving in open spaces or along the same road multiple times where scenery does not change much.

When a new key frame and its features are added into the database, the index of the database has to be dynamically reconstructed for the optimal representation of the new database. However, the computation time on reconstructing the entire index grows linearly with the total number of key frames, which is impractical since the database will keep growing as the vehicle moves. To avoid the cost in reconstructing the entire index, we use collection of trees of exponentially maximal capacities for indexing, such as trees that hold index for features from 4, 16, 64, 256 key frames. When new key frames and their features are added, trees of smaller size are re-built often while large trees are rebuilt rarely by merging smaller trees. In addition, tree merging is performed on a parallel thread to support uninterrupted growth of the database. This way makes the running time of the indexing process to be independent to the number of key frames in the database.

3) Loop Closure Detection: Loop closure detection is also supported during landmark database construction and maintenance. If a query frame is matched to a keyframe that has been added some time ago, the matched keyframe must be acquired when the vehicle previously visited the same place. Therefore, these matches can be treated as loop closures to optimize all poses and landmarks involved within the loop.

C. Inference

In addition to camera feature observations, our system fuses data from IMU and high-precision differential GPS to build a high-quality map of visual landmarks. The inference problem can be easily represented as a factor graph. Figure 3 shows all possible factors from sensor measurements used in either our mapping or navigation system. Note during the mapping process, there are no 2D-3D pre-mapped landmark factors (no orange factors in Figure 3) available.

A factor graph [9] is a bipartite graph model comprising two node types: *factors* $f_i \in \mathcal{F}$ and *state variables* $\theta_j \in \Theta$. Sensor measurements $z_k \in Z$ are formulated into factor representations, depending on how a measurement affects the appropriate state variables. For example, a GPS position measurement only involves a navigation state x at a single time. A camera feature observation can involve both a navigation state x and a state of unknown 3D landmark position l. Estimating both navigation states and landmark positions simultaneously is very popular in SLAM problem formulation, which is also known as bundle adjustment [20] in computer vision. The modeling of IMU data is more complicated, which we will describe in Section IV-A. The

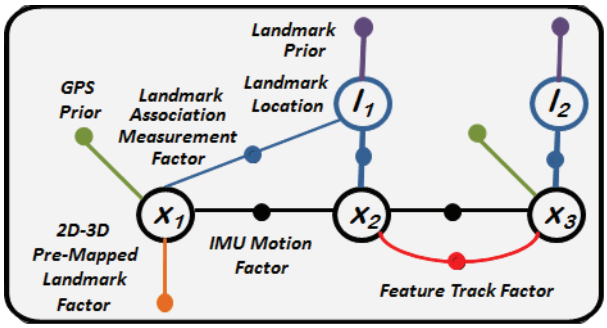

Fig. 3: Factor graph comprising three navigation states and two landmark states. Factors are formed using measurements from GPS, IMU, feature tracks, and pre-mapped landmark observations. Note factors formed from different kinds of sensor measurements are shown as different colors.

inference process of such a factor graph can be viewed as minimizing the non-linear cost function as follows.

$$\sum_{k=1}^{K} ||h_k(\Theta_{j_k}) - \tilde{z_k}||^2_{\Sigma} \tag{1}$$

where $h(\Theta)$ is measurement function and and $||\cdot||^2_{\Sigma}$ is the Mahalanobis distance with covariance Σ.

To solve this inference problem for our mapping system, we take the same approach as [3] which splits the estimation into a fast short-term smoother and a slower long-term smoother. The short-term smoother reconstructs 3D landmarks using tracked visual features from visual odometry module, and provides initial estimates with low-latency to the long-term smoother. The long-term smoother, which keeps all past states, processes expensive loop closures identified from the landmark matching module. This way supports our real-time mapping pipeline by fully optimizing the map of 2D-3D landmarks using both smoothers.

IV. High-Precision GPS-Denied Navigation

Our navigation system is able to achieve high-precision GPS-denied navigation accuracy using pre-mapped visual landmarks in large-scale environments. Figure 4 shows the architecture of our GPS-denied navigation system. Compared to our mapping system (Figure 2), it is simplified for pure localization, without the need for simultaneous map construction. The visual odometry module still tracks features from previous frame to current frame, and provides feature track measurements to the inference engine for tightly-coupled visual-inertial fusion. For new 2D features on the input frame from visual odometry module, the landmark matching module searches the pre-built key-frame database with 2D-3D pre-mapped landmarks, and checks whether there are feature associations between the new video frame and past key frames. The identified associations are treated as pre-mapped landmark observations, which provide global corrections for the inference engine.

A. Tightly-Coupled Visual-Inertial Odometry

Both our navigation system and mapping pipeline (Section III) rely on a tightly-coupled approach [14] to fuse IMU

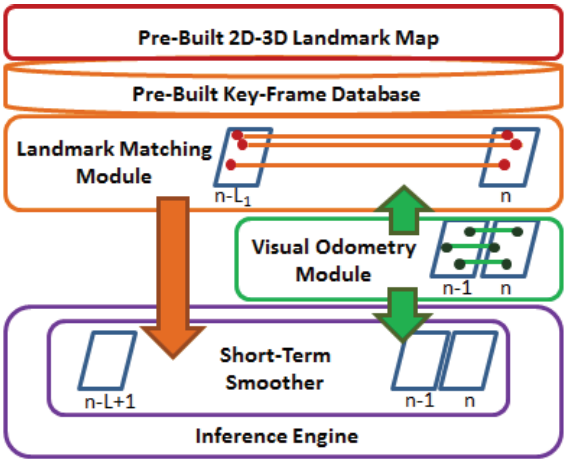

Fig. 4: The architecture of our GPS-denied navigation system. The inference engine involves only the short-term smoother for pure localization problem. The visual odometry module tracks monocular features from previous frame to current frame for the short-term smoother. The landmark matching module matches 2D features from new images to pre-mapped 2D-3D landmarks, and generate associations to the short-term smoother.

data and monocular feature track measurements. Inertial measurements from IMU are produced at a much higher rate than cameras. To fully utilize high-frequency IMU data, we implement a single binary factor (black factors in Figure 3) to summarize multiple consecutive inertial measurements between two navigation states created at the time when other sensor measurements come (such as features from a video frame). This IMU factor generates 6 degrees of freedom relative pose and corresponding velocity change as the motion model. It provides scale information to estimate 3D positions of landmarks from monocular visual features. It also tracks the IMU-specific bias as part of the state variables for estimating motion. We use it instead of traditional process models in both our mapping and navigation systems. In contrast to traditional filtering techniques, this IMU motion factor is part of our non-linear optimization process. The value of IMU integration changes during re-linearization for iterative optimization.

B. Outlier Rejection

Handling faulty visual measurements or association errors to pedestrians or other moving cars is an important topic for vehicle navigation system. These outliers negatively affect navigation accuracy and mapping quality. To improve the robustness of our system, we use a three-layer mechanism to remove these outliers. The first layer is in the visual odometry module (Section III-A). It uses pairwise epipolar constraints across three consecutive frames to discard outlier tracked features, which are re-projected behind cameras or unrealistically close to cameras. Underlying epipolar matrices are estimated using preemptive RANSAC [17] using 2.5 pixel image re-projection error as an inlier measure.

The second layer extends the tightly-coupled visual-inertial approach in Section IV-A to improve the quality of camera feature tracking process. We utilize IMU propagation mechanism, which predicts accurate motion during the short time period across two video frames, to guide the feature matching process and to verify the hypothesis generation through the RANSAC process. This way we are able to generate more inlier features tracked for longer time and distance, which provide better estimates of the underlying 3D landmarks of these tracked features.

The third layer leverages our inference engine (Figure 2 and Figure 4), which stores 3D estimates of current and past tracked features from the visual odometry module. Coupled with geometric constraints, these 3D estimates can be used to prune outliers among new feature observations from the visual odometry module and key frame candidates from the landmark matching module (Section III-B).

C. Pre-Mapped Visual Landmark Observations

Here we describe how we model the pre-mapped visual landmark observations returned from the landmark matching module. Each pre-mapped visual landmark observation represents a correspondence between a new 2D feature from the current video frame and a pre-mapped 2D-3D landmarks in the pre-built map.

We define the navigation state for a given time as $x = \{\Pi, v, b\}$ (Figure 3). Each state x covers three blocks: pose block Π includes 3d translation t (body in global) and 3d rotation R (global to body), velocity block v represents 3d velocity, and b denotes sensor-specific bias. To simplify the notation, we assume all sensors have the same center, which is the origin of the body coordinate system.

Since 3D estimates of pre-mapped landmarks are already fully optimized, the uncertainty of the landmarks is converged and small. We therefore treat the estimated 3D location of a pre-mapped landmark as a fixed quantity (3D point) with the estimated uncertainty from the map in the global coordinate system. This fixed 3D point q is transformed to the body coordinate system as $\mathbf{m} = [\ m_1 \quad m_2 \quad m_3\ ]^T$, based on rotation R_j and translation t_j in state X_j to generate a unary factor (orange factor in Figure 3) with the following measurement model (inspired from [18]).

$$z = h(X_j) + n\ = f(\mathbf{m}) + n = f(R_j(q - t_j)) + n \quad (2)$$

$$f(\mathbf{m}) = f\left([m_1, m_2, m_3]\right) = [\frac{m_1}{m_3}, \frac{m_2}{m_3}] \quad (3)$$

where n is the noise and $f(\mathbf{m})$ is the function that projects m into the normalized 2D imaged point. The Jacobian of the measurement z with respect to R_j and t_j from X_j is calculated as follows:

$$\delta z \simeq G\delta X_j \quad (4)$$

$$G = M([R_j(q - t_j)]_x \delta R_j - R_j \delta t_j) \quad (5)$$

$$M = \begin{bmatrix} \frac{1}{m_3} & 0 & \frac{-m_1}{m_3^2} \\ 0 & \frac{1}{m_3} & \frac{-m_2}{m_3^2} \end{bmatrix} \quad (6)$$

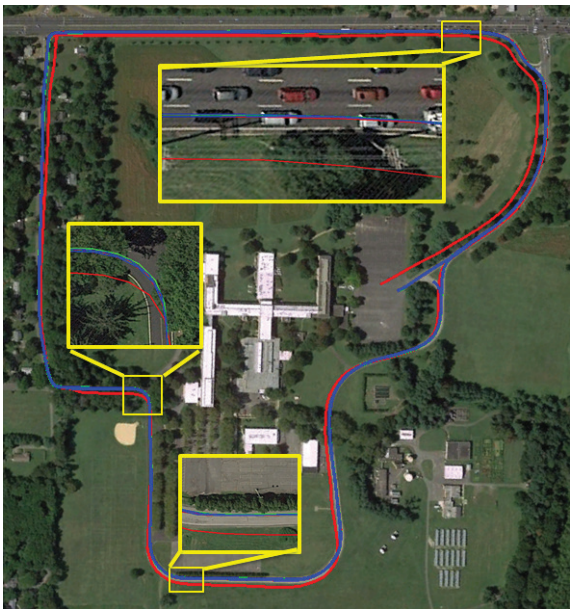

Fig. 5: Our GPS-denied navigation results around our campus: Ground truth (green), navigation without pre-mapped landmarks (red), navigation with pre-mapped landmarks (blue).

Note the above factor formulation is applied to all pre-mapped landmark observations which are identified through feature correspondences between the stored key frame and the current frame. These unary factors (orange factors in Figure 3) provide immediate absolute information to update estimation.

D. Inference

The inference problem of our GPS-denied navigation system (no green factors for GPS measurements in Figure 3) is simplified as pure localization process. We use only the short-term smoother (Figure 4) for inference, which supports non-linear optimization over a sliding constant-length window and efficiently removes states that are outside of the window. It achieves a more optimal solution in real time than traditional filtering methods by checking consistency across a larger collection of sensor measurements.

Note we divide the pre-built map into many sub-maps stored in external disks. The system preserves only the sub-map, which covers current estimated position, in memory during navigation. This way makes our real-time inference process scales well for large-scale navigation.

V. Experimental Results

We validated our system on data we collected across seasons within large-scale urban environments which include a variety of buildings, downtown traffic, highway driving, and lighting variations. The vehicle we used for experiments incorporates a 100Hz MEMS Microstrain GX4 IMU and one 20Hz front-facing monocular Point Grey camera. High-precision differential GPS, which is also installed on the vehicle, was used both for geo-referenced map construction and for ground truth generation (fused with IMU) to evaluate our GPS-denied navigation system. All three sensors are calibrated and triggered through hardware synchronization. Note we are not aware of any publicly available vehicle data that provides raw IMU and differential GPS measurements we need at same places across seasons.

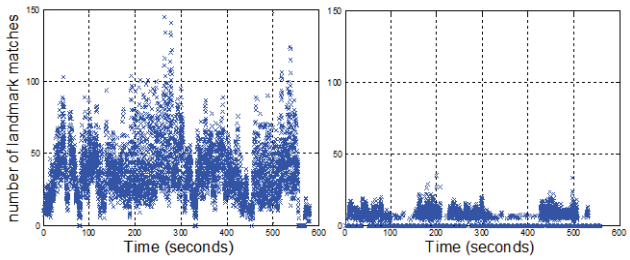

Fig. 6: Number of visual landmark matches (left: winter, cloudy noon, right: spring, sunny morning) during navigation.

A. Campus Driving

We first performed experiments around our campus, which include highway (top portion in Figure 1) and roads near trees. The map of visual landmarks was constructed during data collection driving on a winter morning. It includes 232495 2D-3D landmarks with 588 key frames.

1) Improvement from pre-mapped landmarks: We then tested our GPS-denied navigation system using this map by driving two continuous loops around our campus in clockwise direction on a cloudy winter noon. The total driving distance is 5.6 km and the total driving time is 582 seconds. Figure 5 shows our results and ground truth trajectory (green). For comparison, we've also generated GPS-denied navigation result without pre-mapped landmarks (red), which relies on only IMU data and local feature track measurements. It drifts more during the second loop and eventually achieves 3D RMS error in 7.9602 meters, which is reasonable for 5.6 km driving without any absolute measurements. However, by utilizing pre-mapped visual landmarks, our GPS-denied navigation result (blue) reaches sub-meter accuracy all the way: the 3D RMS error is only 0.5378 meters while the 90 percentile 3D error is 0.7878 meters. As shown from the enlarged regions in Figure 5, the trajectory is nearly identical to ground truth and sticks on correct lane of road all the time. The average lateral error is 0.2212 meters, and the average longitudinal error is 0.2272 meters.

2) Robustness to scene changes : To demonstrate the robustness of our navigation approach to scene changes, we tested our system using the map built in winter for two GPS-denied navigation runs (same route in section V-A.1) in spring under different weathers (cloudy afternoon, sunny morning). Figure 6 shows the number of landmark observations decreases from winter to spring (at most 35 matches/frame in spring). Visual appearance of same places becomes very different due to changes in lighting, shading, and scene objects (such as cars and trees), as shown in the three examples in Figure 7. Only permanent features from man-made structures such as houses and far tree scanlines remain reliable.

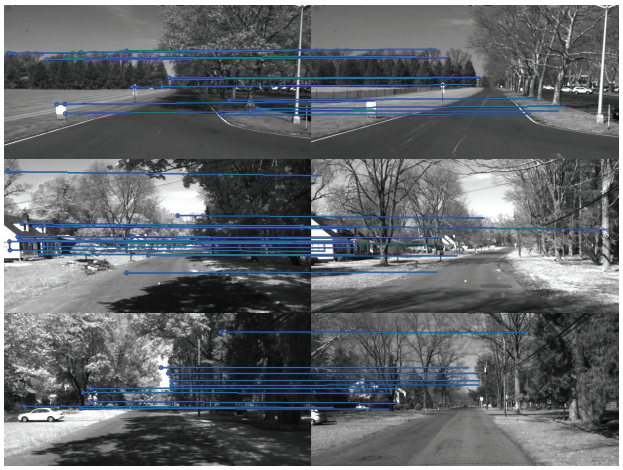

Fig. 7: Three examples of 2D feature matches between features observed in spring (left) and landmarks mapped in winter (right).

The first example (top row in Figure 7) shows the easier condition, where trees grew leaves and fence is removed from the left side of the road. Despite that, clear view of far tree scanline and presence of distinct road signs allows relatively effortless matching. The other two examples are notoriously more challenging for landmark matching. Images look very different due to different sun position, which creates drastically different shading and lighting, as well as more tree vegetation on left images taken in spring. Furthermore, some objects are missing (the tree on the left part of the foreground in the second example) or newly introduced (the car in the left part of the foreground in the third example). Therefore, only relatively constant features from the houses, poles and far trees scanline are capable to yield reliable matches to improve navigation performance.

To increase the robustness to scene changes, where only few or no mapped landmarks are observed for global corrections, our method tightly incorporates IMU data and relative feature track measurements to propagate 3D motion estimation. This way has been shown to propagate accurate 3D pose for longer periods [11] than loosely-coupled approach [10]. Using our approach, 3D RMS error for these two tests are 0.9440 meters (spring cloudy afternoon) and 1.1200 meters (spring sunny morning) respectively.

3) Improvement from point measurement model: Note our navigation method treats each observation of a pre-mapped landmark as a single measurement, instead of computing only one pose measurement [1], [10] from all landmark observations at a given time. This way allows us to model individual landmarks with different uncertainties, and tightly incorporates absolute geo-referenced information into these measurements.

To validate the influence from our individual point modeling of single pre-mapped landmarks, we implemented and tested pose measurement from [1], [10] for comparison. Note we use same GPS-denied navigation pipeline for experiments. The only difference is the way to fuse observations of pre-mapped landmarks. 3D RMS error on three test sequences using pose measurements is 1.0434 meters (winter, cloudy noon), 1.5813 meters (spring, cloudy afternoon), and 1.8940 meters (spring, sunny morning) respectively. Fusing with our point measurements reduces the error by 40% to 50%. The 3D RMS error becomes 0.5378 meters, 0.9440 meters, and 1.1200 meters respectively.

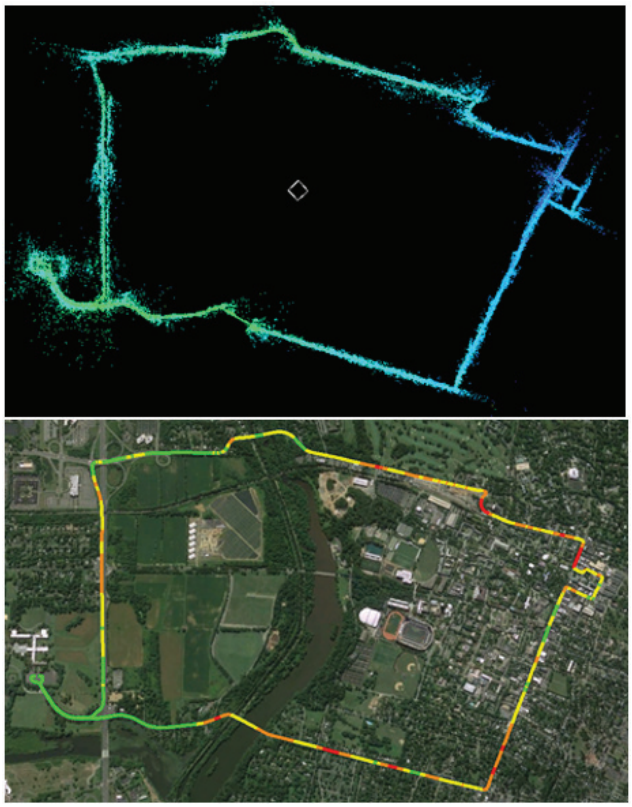

Fig. 8: (top) 3D visual landmarks mapped around a city in real time. The color code reflects the absolute height. The small white box shows the 3D perspective viewpoint. (bottom) 3D error E in meters with color representation along the navigation trajectory : $E < 0.5$ in green, $0.5 < E < 1.0$ in yellow, $1.0 < E < 1.5$ in orange, and $1.5 < E < 2.0$ in red.

B. City Driving

We also conducted experiments within a city, including downtown streets with traffic. Figure 8 (top) shows 3D point cloud of visual landmarks which are pre-mapped within the city. It includes 471419 high-quality 2D-3D landmarks with 1018 key frames. Using this map, our vehicle drove along the same route around the city without GPS. The total driving distance is 10.14 km and the total driving time is 1292 seconds. 3D RMS error of the entire trajectory is 0.9365 meters, which achieves sub-meter navigation accuracy.

To further investigate the performance, the varied 3D error along the trajectory estimated from our system using pre-mapped visual landmarks is represented with different colors (bottom of Figure 8). Note the right side of the map is city downtown. There is traffic at road interactions in regions at both upper-right and bottom-right corners, which results in higher 3D error. The 3D error is less than 0.5 meters when the vehicle drove to top-left and bottom-left regions, which have less traffic than downtown streets.

Fig. 9: (top) Ground truth (green), non-differential GPS (red), and GPS-denied navigation with pre-mapped landmarks (blue). (bottom) Examples of camera frames perceived during navigation.

Figure 9 shows an enlarged portion of our GPS-denied navigation results on downtown streets, with examples of video camera frames captured during driving. When our test vehicle drove on downtown streets or road interactions, the camera view is easily blocked (partially or totally) by other vehicles, if our vehicle is close to front cars. However, our GPS-denied navigation trajectory with pre-mapped landmarks (blue) is still very close to ground truth (green), and is located on the correct lane of streets. For comparison, our vehicle is also equipped with a non-differential GPS system, which performs poorly (red) in downtown city due to signal obstructions caused by urban street canyons.

VI. Conclusion

In this paper, we present our vehicle navigation system which achieves sub-meter GPS-denied accuracy in large-scale urban environments using pre-mapped visual landmarks. Our navigation system tightly couples IMU data with local feature track measurements, and treats each new observation of a pre-mapped visual landmark as a single global measurement. This approach propagates precise 3D global pose estimates for longer periods in GPS-denied setting, and is more robust when only few or no pre-mapped landmarks are available due to scene occlusion or changes.

To construct a high-quality map of visual landmarks beforehand, our mapping pipeline adopts a dual-layer architecture to fuse measurements from a monocular camera, IMU, and high-precision differential GPS. This architecture utilizes low-latency navigation updates to provide an initial estimate for slower map optimization running in parallel, and is able to construct the fully optimized map in real time.

Future work is to enhance the visual map construction by intelligently gathering data from multiple collections. For example, permanent objects and objects such as parked cars can be automatically separated in the map. We believe our system can serve as the basis for a robust, efficient strategy for GPS-denied navigation with centimeter-level accuracy.

Acknowledgments

This material is partially based upon work supported by the DARPA All Source Positioning and Navigation (ASPN) Program under USAF/ AFMC AFRL Contract FA8650-13-C-7322. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the US government/ Department of Defense.

References

[1] C. Beall and F. Dellaert. Appearance-based localization across seasons in a metric map. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) workshops*, 2014.

[2] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: binary robust independent elementary features. *European Conference on Computer Vision (ECCV)*, 2010.

[3] H. Chiu, S. Williams, F. Dellaert, S. Samarasekera, and R. Kumar. Robust vision-aided navigation using sliding-window factor graphs. *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2013.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[5] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics and Automation Magazine*, 13(2), 2006.

[6] E. Guizzo. How google's self-driving car works. *IEEE Spectrum*, 2011.

[7] C. Harris and M. Stephens. A combined corner and edge detector. *Proc. of Fourth Alvey Vision Conference*, 1988.

[8] M. Kaess, S. Williams, V. Indelman, R. Roberts, J. Leonard, and F. Dellaert. Concurrent filtering and smoothing. *International Conference on Information Fusion*, 2012.

[9] F. Kschischang, B. Fey, and H. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. Information Theory*, 47(2), 2001.

[10] H. Lategahn, M. Schreiber, J. Ziegler, and C. Stiller. Urban localization with camera and inertial measurement unit. *IEEE Intelligent Vehicles Symposium (IV)*, 2013.

[11] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale. Keyframe-based visual-inertial odometry using non-linear optimization. *Intl. J. of Robotics Research*, 34(3):314–334, 2015.

[12] J. Levinson and S. Thurn. Robust vehicle localization in urban environments using probabilistic maps. *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2011.

[13] Y. Liu and H. Zhang. Indexing visual features: real-time loop closure detection using a tree structure. *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2012.

[14] T. Lupton and S. Sukkarieh. Visual-inertial-aided navigation for high-dynamic motion in built unvironments without initial conditions. *IEEE Trans. Robotics*, 28(1):61–75, 2012.

[15] M. Muja and D. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. *Intl. Conf. on Computer Vision Theory and Applications (VISAPP)*, 2009.

[16] R. Newcombe, S. Lovegrove, and A. Davison. DTAM: Dense tracking and mapping in real-time. *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[17] D. Nister. Preemptive ransac for live structure and motion estimation. *IEEE Intl. Conf. on Computer Vision (ICCV)*, 2003.

[18] T. Oskiper, H. Chiu, Z. Zhu, S. Samarasekera, and R. Kumar. Stable vision-aided navigation for large-area augmented reality. *IEEE Intl. Conf. on Virtual Reality (VR)*, 2011.

[19] S. Song, M. Chandraker, and C. Guest. Parallel, real-time monocular visual odometry. *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2013.

[20] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment - a modern synthesis. *Lecture Notes in Computer Science*, 1883:298–375, 2000.

[21] H. Uchiyama, D. Deguchi, T. Takahashi, I. Ide, and H. Murase. Ego-localization using streetscape image sequences from in-vehicle cameras. *IEEE Intelligent Vehicles Symposium (IV)*, 2009.

[22] D. Wong, D. Deguchi, I. Ide, and H. Murase. Vision-based vehicle localization using a visual street map with embedded surf scale. *European Conference on Computer Vision (ECCV) workshops*, 2014.